VŠB - Technická univerzita Ostrava

Fakulta elektrotechniky a informatiky

Katedra telekomunikační techniky

DIPLOMOVÁ PRÁCE

2010 **Bc. Petr Adámek**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Detekce a prevence průniků do síťových systémů

Network intrusion detection and prevention systems

2010 **Bc. Petr Adámek**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Zadání diplomové práce

Student: **Petr Adámek**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2601T013 Telekomunikační technika

Téma: Detekce a prevence průniků do síťových systémů
Network intrusion detection and prevention systems

Zásady pro vypracování:

Cílem diplomové práce je navrhnout a ověřit vhodné zabezpečení sítě pomocí open source nástroje SNORT.
1. Úvod do problematiky IDS/IPS.
2. Návrh a implementace detekčního systému.
3. Ověření funkčnosti pomocí různých penetračních nástrojů.

Seznam doporučené odborné literatury:

Podle pokynů vedoucího diplomové práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Pavel Nevlud**

Datum zadání: 20.11.2009
Datum odevzdání: 07.05.2010

VYSOKÁ ŠKOLA BÁŇSKÁ-TECHNICKÁ UNIVERZITA Fakulta elektrotechniky a informatiky OSTRAVA

prof. RNDr. Vladimír Vašinek, CSc.
vedoucí katedry

prof. Ing. Ivo Vondrák, CSc.
děkan fakulty

Čestné prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

V Ostravě 7. 5. 2010

……

Chtěl bych především poděkovat panu Ing. Pavlu Nevludovi za neocenitelnou podporu, poskytnutí materiálů, a také za odborné vedení mé diplomové práce. Dále bych chtěl poděkovat své přítelkyni za psychickou podporu a gramatickou korekturu.

ABSTRAKT

Na začátku mé diplomové práce je objasněna problematika IPS/IDS systémů, jejich typy a možnosti nasazení v praxi. Další část je zaměřena na IDS systém Snort, jeho vlastnosti a nastavení pro účinnou detekci průniků v síti. V poslední části jsou objasněny pojmy týkající se penetračního testování a následně pomocí různých bezpečnostních nástrojů je otestován navržený systém**.**

KLÍČOVÁ SLOVA

IDS, IPS, IDSwakeup, Nessus, Nmap, Snort

ABSTRACT

At the beginning of my diploma thesis IPS/IDS systems, their types and possible deployment strategies are explained. The next section is focused on IDS system Snort, its features and set-up for effective detection of network intrusions. In the last section, terms relating to penetration testing are clarified and proposed system is tested by using various security tools.

KEYWORDS

IDS, IPS, IDSwakeup, Nessus, Nmap, Snort

SEZNAM POUŽITÝCH SYMBOLŮ A ZKRATEK

DIDS (Distributed Intrusion Detection System) - distribuované systémy detekce narušení

DoS (Denial of Service) - odepření služby

HIDS (Host Based Intrusion Detection Systems) - uzlově orientované systémy detekce narušení

ICMP (Internet Control Message Protocol) - protokol pro přenos chybových a řídících zpráv

IDS (Intrusion Detection System) - systémy pro detekci útoků

IP (Internet Protocol) - protokol používaný pro přenos dat

IPS (Intrusion Prevention System) - systémy ochrany proti průniku

MAC (Media Access Control) - podvrstva druhé vrstvy ISO/OSI

NIC (Network Interface Controller) - síťová karta

NIDS (Network Based Intrusion Detection Systems) - síťově orientované systémy detekce narušení

TCP (Transmission Control Protocol) - spojově orientovaný protokol

TOS (Type of service) - typ služby

TTL (Time to Live) - doba života

UDP (User Datagram Protocol) - nespojově orientovaný protokol

Obsah

1 Úvod

V dnešním světě plném vyspělé elektroniky a internetu je téměř samozřejmostí a nutností práce s počítačem. Dnes má skoro každá domácnost počítač připojený k internetu. Tato skutečnost nám umožňuje získávat velké množství informací, jak z domácích, tak i ze zahraničních serverů. S touto výhodou přichází i fakt, že většina lidí o počítačích a jejich zabezpečení moc neví. Proto je nutné používat na osobních počítačích antivirový software, osobní firewall a pravidelně je aktualizovat. Důležitou součástí je také pravidelná aktualizace operačního systému. V rámci podnikových sítí je vhodné nasadit IDS/IPS systémy, které detekují neautorizované síťové aktivity a dokáží jim zabránit.

Jako téma své diplomové práce jsem si vybral detekci a prevenci útoků do síťových systémů, jelikož mě oblast bezpečnosti zajímá. Práce je členěna do třech základních částí. V první části je objasněna problematika IDS/IPS systémů. Jsou zmíněny všechny hlavní typy těchto systémů, jejich výhody a nevýhody. V druhé části se zabývám síťovým IDS systémem Snort, jeho vlastnostmi, pravidly, a v neposlední řadě taky nastavením pro účinnou detekci útoků. Poslední část je zaměřena na testování navrženého zapojení různými penetračními nástroji. Je využit skener portů, nástroj pro kompletní otestování IDS systémů a také bezpečnostní skener, sloužící k analýze zranitelnosti systémů.

Součástí diplomové práce je i CD, na kterém jsou uloženy všechny generované výstrahy, kompletní logy a konfigurační soubory.

2 IDS/IPS systémy

2.1 IDS systémy

IDS nebo-li (Intrusion Detection Systems) jsou systémy, které identifikují neautorizované síťové aktivity. Lépe řečeno identifikují aktivity které mohou, ale také nemusí být narušeními. Tyto systémy mohou být jak softwarovým, tak hardwarovým řešením či kombinací obou. Detekce narušení bývá pouze jednou z části obranného sytému. Zpravidla je instalována na nějakém zařízení či systému. Pro detekci narušení a jeho prevenci pracují IDS systémy většinou ve spolupráci s IPS systémy a firewally. IDS pracují na síťové vrstvě OSI modelu. Pasivní síťové senzory bývají umísťovány do tzv. choke pointů. Tady probíhá analýza paketů, zda neobsahují typické vzorky neautorizovaných aktivit. Pokud je takový vzorek nalezen, je zaznamenána výstraha. Na jejím základě mohou být učiněna odpovídající opatření.

Systémy IDS se velice podobají antivirovému softwaru. Obsahují známé signatury (podobně jako virové definice u antivirů), podle nichž dokáží detekovat potencionálně nebezpečné síťové aktivity. IDS systémy umožňují detekci těchto aktivit na základě pravidel a na základě detekce anomálií.

- *Detekce dle pravidel* - systém hledá známé vzorky obsažené v příchozích paketech a porovnává je s databází pravidel obsažených v IDS systému.
- *Detekce na základě anomálií* - systém analyzuje tok paketů a podle něj se snaží detekovat odchylky oproti standardnímu toku dat. [2]

2.2 Typy IDS systémů

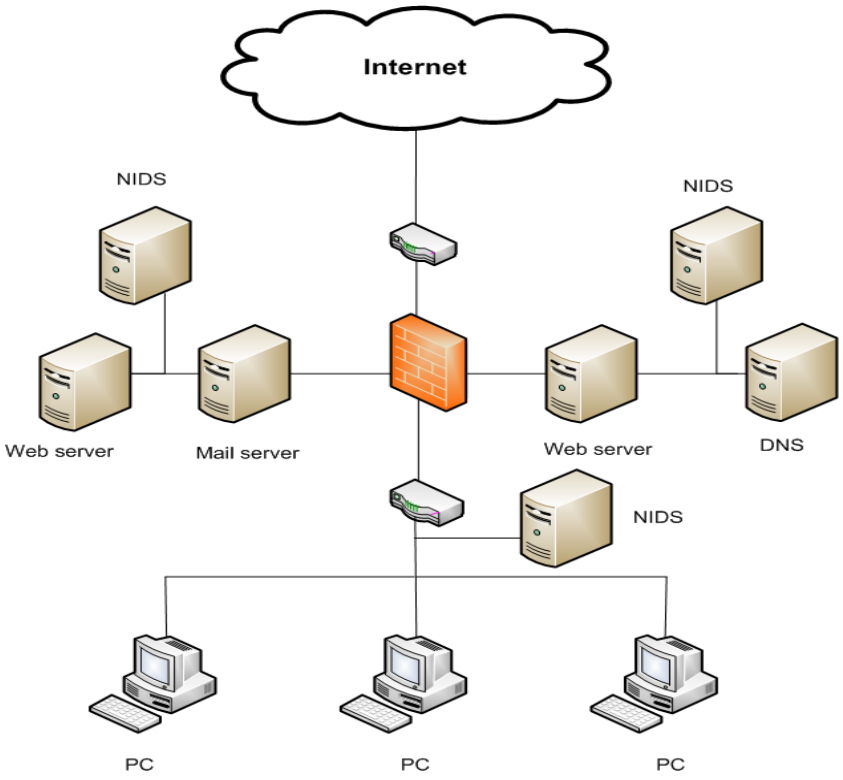
IDS systémy jsou rozděleny do následujících tří skupin:

- NIDS (Network Based Intrusion Detection Systems) - síťově orientované systémy detekce narušení
- HIDS (Host Based Intrusion Detection Systems) - uzlově orientované systémy detekce narušení
- DIDS (Distributed Intrusion Detection System) - distribuované systémy detekce narušení

2.2.1 NIDS

Tyto systémy detekce narušení přijímají a analyzují pakety v jednotlivých segmentech sítě. Běžně pracují síťové karty (NIC-Network Interface Card) v nepromiskuitním módu, což znamená, že pro analýzu provozu jsou předávány pouze pakety určené pro MAC (Media Access Control) této síťové karty. Aby mohly NIDS systémy mohly monitorovat provoz na síti, musí pracovat v tzv. promiskuitním módu, který není omezen zachytáváním paketů pouze z jediné síťové karty, ale umožňuje zachytávat celý síťový provoz.

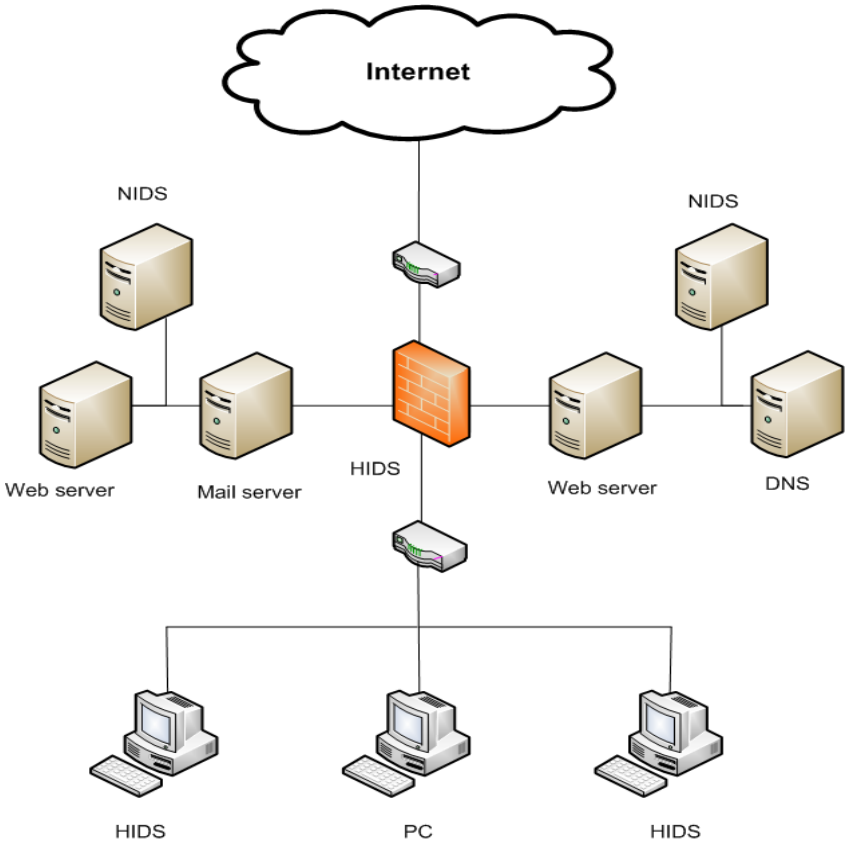
Obrázek 1 zobrazuje síť, ve které jsou použity tři NIDS. Tyto jednotky byly umístěny na strategických segmentech sítě, a monitorují provoz pro všechny zařízení umístěné v daném segmentu. Použití více NIDS v rámci sítě je příkladem tzv. *hloubkové bezpečnostní architektury*. Mezi její výhody patří schopnost monitorovat rozsáhlou síť, přičemž provoz na není nikterak ovlivňován. Nevýhody spočívají ve zpracování paketů při velkém zatížení sítě a v nemožnosti analýzy šifrovaného provozu. [4]



Obr. 1: NIDS síť [4]

2.2.2 HIDS

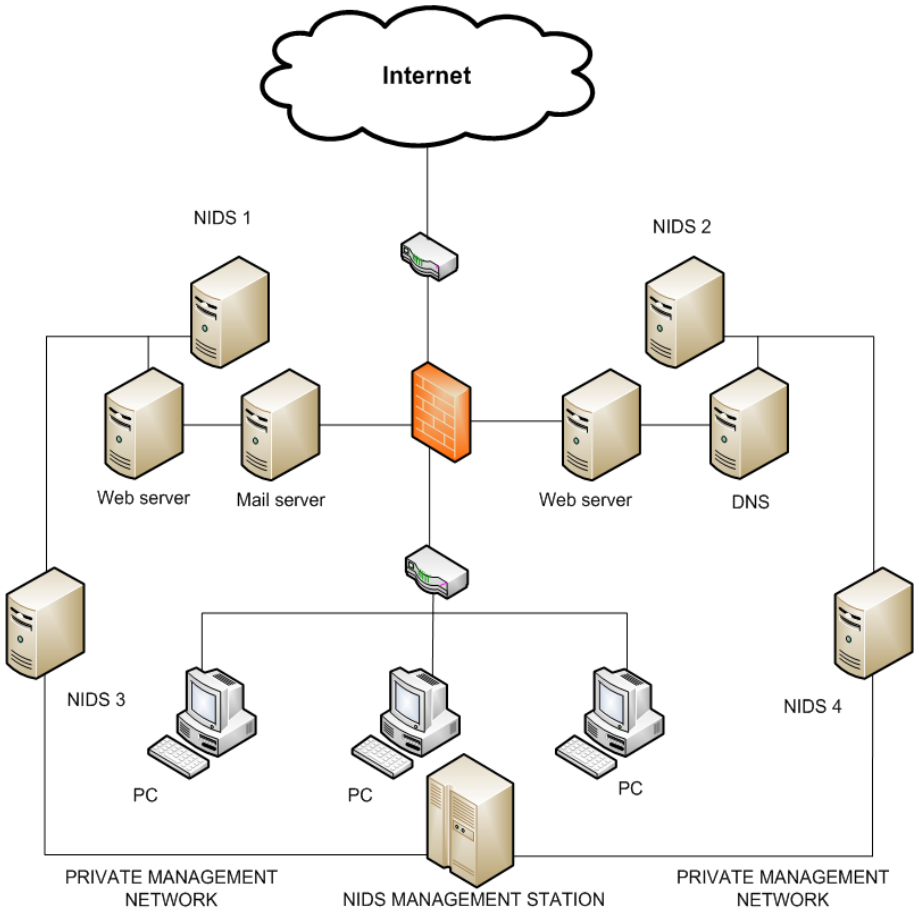
Systémy tohoto typu se liší od systémů NIDS dvěma způsoby. HIDS systémy chrání pouze hostitelský systém, na kterém jsou nainstalovány. Jeho síťová karta pracuje ve výchozím nastavení v nepromiskuitním módu. Tento mód může být výhodou pouze v některých případech. Ne všechny síťové adaptéry jsou schopny promiskuitního módu. Kromě toho, promiskuitní režim je náročný na procesor, což může omezovat použití na pomalých hostitelských počítačích. Další výhodou těchto systémů je schopnost velmi podrobně přizpůsobovat pravidla pro každý hostitelský počítač. V důsledku snížení počtu relevantních pravidel se zvyšuje výkon procesoru a snižuje potřebný výpočetní výkon. V průběhu instalace detekčního systému na jednotlivých hostitelských strojích může být nakonfigurován společný soubor pravidel. Obrázek 2 zobrazuje síť pomocí HIDS umístěných na konkrétních serverech a hostitelských počítačích. Umí analyzovat šifrovaný provoz. Mezi nevýhody patří složitá administrace, možnost vyřazení z provozu útokem typu DoS. [4]



Obr. 2: HIDS síť [4]

2.2.3 DIDS

Tyto systémy mají architekturu tzv. manažer/sonda, z čehož vyplývá, že jednotlivé sondy reportují v určitých časových intervalech dění na centrální stanici (management station). Senzory umístěné na síti jsou tedy spravovány centrálně a mohou být typu NIDS, HIDS nebo kombinací obou. Díky tomu mohou jejich síťové karty pracovat jak v promiskuitním, tak i v nepromiskuitním režimu. [4]



Obr. 3: DIDS síť [4]

2.3 IPS systémy

Systémy IPS (Intrusion Prevention Systems), nebo-li systémy ochrany proti průniku. Jsou velmi podobné systémům IDS. Umožňují detekci útoků, jsou schopny na ně reagovat (zabránit útoku nebo ho přerušit).

Existují dva druhy IPS systémů:

2.3.1 HIPS (Host IPS)

Spoléhají se na agenty nainstalované přímo v chráněném systému. Vzhledem k tomu, že HIPS agent zachytává všechny požadavky na systém, který chrání, musí být velmi spolehlivý, nesmí negativně ovlivňovat výkon, ani blokovat běžný provoz.

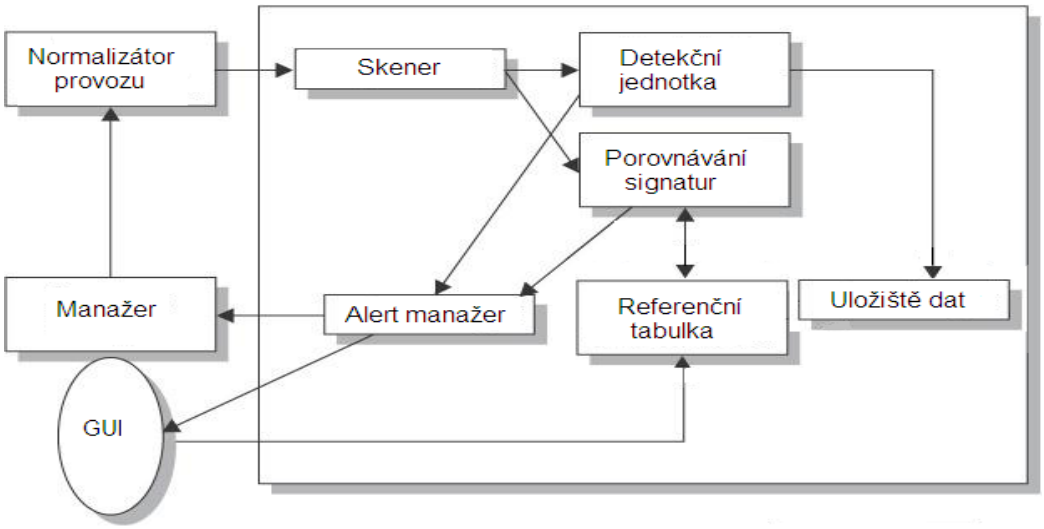
2.3.2 NIPS (Network IPS)

Tento systém v sobě kombinuje prvky klasického IDS, IPS systému a firewallu, říká se mu taky In-line IDS nebo Gateway IDS. Stejně jako u firewallů, NIPS má nejméně dvě síťová rozhraní, jedno označeno jako interní, jedno jako externí. Příchozí pakety se objevují na obou rozhraních, dále jsou pak předávány detekční jednotce, která určuje, zda paket představuje hrozbu či ne.

Typické IPS se skládá ze čtyř komponent:

- Normalizátor provozu
- Monitor služeb
- Detekční jednotka
- Tvarovač

Normalizátor provozu analyzuje síťový provoz a plní blokovací funkci. Poté je signál předáván do detekční jednotky a do monitoru služeb. Ten informaci oklasifikuje a pomáhá tvarovači při řízení toku dat. Detekční jednotka má za úkol porovnávat přijaté vzory s referenční tabulkou. [2]



Obr. 4: Standartní IPS systém [2]

2.4 Srovnání IDS a IPS systémů

2.4.1 IDS systémy

Výhody

- Nabízí centrální správu
- Jsou pasivním prvkem v síti
- Umožňuje detekci jak z vnitřní, tak z vnější sítě
- Poskytují hloubkovou analýzu

Nevýhody

- Nejsou schopny analyzovat šifrovaný provoz
- Nepředchází útoku, pouze vydává výstrahu
- Generují velké množství dat

2.4.2 IPS systémy

Výhody

- Zabezpečují aplikační vrstvu
- Jsou aktivním prvkem v síti
- Umožňují předcházení útoků
- Nabízí centrální správu

Nevýhody

- Generují falešné alerty
- Je to nová a nákladná technologie
- Vytváří úzká místa v síti [2]

3 Architektura systémů IDS/IPS

Architektura je jedním z nejkritičtějších aspektů v nasazení detekce a prevence narušení. Její špatný návrh vede k několika nežádoucím jevům, kterými jsou nedostupnost dat, špatná odezva, či zpomalení sítě.

3.1 Vrstvená architektura

Je složena ze tří typů:

- jednovrstvá architektura
- vícevrstvá architektura
- architektura peer-peer.

3.1.1 Jednovrstvá architektura

Je to jedna z nejzákladnějších architektur. Komponenty sbírají a zpracovávají data samostatně, aniž by je předávaly jako svůj výstup jiným komponentám. Tato architektura je jednoduchá, levná, a nezávislá. V dnešní době se většinou skládá z komponent, které o sobě vzájemně neví, z čehož vyplývá omezení účinnosti. [2]

3.1.2 Vícevrstvá architektura

Skládá se z více komponent, které si mezi sebou předávají informace. Většina dnešních IDS systémů obsahuje tři základní komponenty jimiž jsou senzory, analyzátory nebo agenty a manažer. Senzory jsou komponenty, které sbírají data ze síťového rozhraní, logů či firewallů. Agenti získávají informace ze senzorů a sledují aktivitu narušení v místě nasazení. Většinou se zaměřují na vykonávání jediné funkce, tj. jeden agent monitoruje pouze TCP spojení, druhý jen UDP spojení atd. Pokud zjistí, že dochází k útoku, pošle informaci komponentě manažer, která má možnost zachovat se podle následujících možností:

- Sběr výstah
- Spuštění stránkovače či vytočení telefonního čísla
- Uložení informace do databáze
- Získávání dalších informací týkající se události
- Zaslání informace na host
- Zaslání příkazů na firewall, pro změnu přístupového kontrolního listu
- Poskytnutí uživatelského rozhraní řídící konzoli

Mezi výhody vícevrstvové architektury patří vyšší účinnost a hlubší analýza. Každá komponenta provádí pouze tu úlohu, pro kterou byla navržena, navíc není závislá na ostatních částech architektury. Hlavní nevýhodou této architektury je složitost, a z toho vyplývající náročnější údržba. [2]

3.1.3 Architektura peer-peer

V této architektuře se vyměňují informace o detekci či prevenci mezi rovnocennými komponentami, z nichž každá provádí stejný druh činnosti. Často se využívá ve spolupráci s firewally. Informace odehrávající se na jednom firewallu se předává druhému, což umožňuje změnu pravidel v řídícím přístupovém seznamu, či přidání restrikcí. Taktéž druhý firewall si vyměňuje informace s prvním, který na základě jeho informací může změnit své chování. Hlavní výhoda spočívá v jednoduchosti. Každý rovnocenný partner může získávat informace od ostatních. [2]

3.2 Senzory

Senzory jsou vlastně vstupními body systému (nejnižšími komponentami), které sbírají a předávají data dalším komponentám. [2]

Existují dva základní typy senzorů:

3.2.1 Senzory založené na síti

Jsou to programy nebo síťová zařízení, která zachytávají data na síti. Ve srovnání se senzory založenými na uzlech bývají nasazovány častěji. Největší výhoda spočívá v počtu uzlů, pro které může senzor poskytovat data. Bude-li mít tato síť sto uzlů, může tento senzor sbírat data o zneužití ve všech uzlech. [2]

3.2.2 Senzory založené na uzlech

Síťové rozhraní každého uzlu musí být nakonfigurováno tak, aby zpracovávalo pouze data určená pro něj. Většina těchto senzorů jsou programy, které vytvářejí data ve formě logu. Výstup těchto programů je zasílán do analytických programů, kde bývá dále zpracováván. [2]

3.3 Agenti

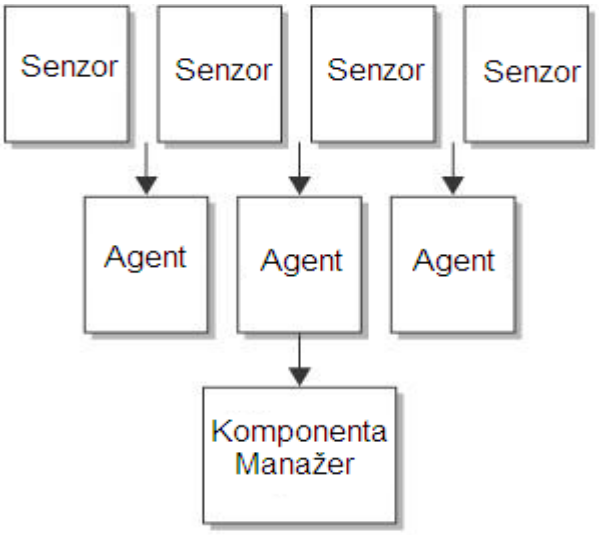
Primární funkcí agentů je analýza dat přijatých od senzorů. Agenta si můžeme představit jako několik na sobě nezávislých procesů analyzující chování, za účelem detekce narušení a anomálií. Pokud se v síti nachází více agentů, pracují nezávisle na sobě. Co z toho plyne? Pokud jeden agent přestane pracovat, ostatní mohou dále pokračovat ve své činnosti. Díky této vlastnosti mohou být jednotliví agenti dle potřeby do systému přidáváni, či odebíráni. Fungují nezávisle na sobě a často spolupracují. [2]

3.4 Komponenta manažer

Někdy také bývá nazývána server, je další komponentou vícevrstvé architektury a zajišťuje funkci řízení IDS/IPS systémů. Manažer je tzv. mozkem celého systému. Jednou z hlavních úloh této komponenty je generování výstrahy, jakmile se objeví událost vysokého stupně nebezpečí. Tyto výstrahy bývají rozesílány e-mailem nebo pomocí syslogu. [2]

Další úlohy komponenty manažer:

- Korelace událostí
- Analýza vyšší úrovně
- Monitorování ostatních komponent
- Generování a distribuce strategie
- Řídící konzole



Obr. 5: Vícevrstvá architektura [2]

4 Návrh a implementace detekčního systému

4.1 Snort

Snort je moderní bezpečnostní nástroj patřící do kategorie síťových IDS systémů. Přicházející provoz filtruje a porovnává ho se svými pravidly, čímž je schopen odhalit případný útok. Je naprogramován v jazyce C, pro záchyt paketů využívá knihovnu libpcap. Snort umí pracovat obdobně jako hardwarový IPS na síťové vrstvě, kdy nemusí mít přiřazenou IP adresu. Podle definovaných pravidel analyzuje obsah paketů. Je schopen detekovat provoz protokolů TCP, UDP, ICMP a IP. [2]

4.1.1 Systémové požadavky

4.1.1.1 Softwarové

Snort může být instalován a provozován na operačních systémech Linux, FreeBSD, NetBSD, OpenBSD, Windows, Solaris, MacOS X a dalších.

4.1.1.2 Hardwarové

Hardwarové požadavky závisí na velikosti monitorované sítě a na provozu, jím probíhající. Minimální požadavky pro ověření funkčnosti mého zapojení jsou procesor taktovaný na 1GHz, pamět RAM o velikosti 1GB, pevný disk o velikosti 20GB, 100Mb/s síťová karta.

4.1.1.3 Software třetích stran

Knihovna libpcap tvoří rozhraní mezi síťovou kartou a programem. Zachycená data z rozhraní jsou předávaná ke zpracování právě této knihovně, která je vyfiltruje a předá dál.

Pokud bychom potřebovali zachytávat data do databáze, bude nutné nainstalovat např. MySQL. Dalším podpůrným softwarem je ACID (Analysis Console for Intrusion Detection), Apache (webový server), Oinkmaster (jednoduchý skript pro správu a aktualizaci pravidel), SnortSnarf (Snort Analyzer), BASE (Basic Analysis and Security Engine). [2]

4.1.2 Režimy Snortu

Snort může pracovat ve třech různých režimech:

- sniffer mode (režim slídiče)
- packet logger (režim záznamu)
- NIDS (režim detekce narušení).

4.1.2.1 Sniffer mode

V tomto režimu čte Snort pakety ze zadaného rozhraní a zobrazuje je na obrazovce. Spuštění programu z příkazové řádky nám umožní tento zápis:

./snort –v

Příklad generovaného varování:

```
03/25-14:37:27.227997 10.0.0.2:67 -> 192.168.1.3:68
UDP TTL:16 TOS:0x10 ID:0 IpLen:20 DgmLen:328
Len: 300
```

Pro zobrazení IP, TCP, UDP a ICMP hlaviček spustíme program příkazem:

./snort –vd

```
03/25-14:50:38.106429 10.0.0.2:5353 -> 192.168.1.3:5353
UDP TTL:255 TOS:0x0 ID:0 IpLen:20 DgmLen:68 DF
Len: 40 00 00 00 00 00 01 00 00 00 00 00 00 0B 5F 70 67
.............._pg 70 6B 65 79 2D 68 6B 70 04 5F 74 63 70 05 6C 6F
pkey-hkp._tcp.lo    63    61    6C    00    00    0C    00    01
cal.....
```

Pokud nás zajímá zobrazení i druhé (linkové vrstvy) paketu, zadáme příkaz:

./snort –vde

Příklad generovaného varování:

```
02/18-14:37:27.196632 0:0:C:46:3E:8B -> 0:2:B3:2B:6B:25
type:0x800 len:0x5D 10.0.0.2:1035 -> 192.168.1.3:23 TCP TTL:63
TOS:0x10 ID:41628 IpLen:20 DgmLen:79 DF ***AP*** Seq: 0x65AA3327
```

```
Ack: 0x62DE8CEB  Win: 0x16D0  TcpLen: 32 TCP Options (3) => NOP
NOP TS: 233919 133491
FF FD 03 FF FB 18 FF FB 1F FF FB 20 FF FB 21 FF  ...........
..!.FB 22 FF FB 27 FF FD 05 FF FB 23                ."..'.....#
```

4.1.2.2 Packet logger mode

V tomto režimu čte Snort pakety ze zadaného rozhraní a ukládá je na disk. Spuštění programu z příkazové řádky nám umožní tento zápis:

./snort -dev –l /var/log/snort

Parametr – *l* říká, že se bude logovat, *var/log/snort* je cesta pro ukládání logu.

Pokud chceme zachytávat data např. z lokální sítě, je možné spustit program s následujícími parametry:

./snort -dev -l ./log -h 192.168.1.0/24

4.1.2.3 Network Intrusion Detection System Mode

Analyzuje provoz na základě pravidel a podle nich generuje varování. Spuštění programu z příkazové řádky nám umožní tento zápis:

snort -c /etc/snort/snort.conf -l /var/log/snort

Přepínač -c říká, že snort poběží v režimu IDS, */etc/snort/snort.conf* nám sděluje cestu ke konfiguračnímu souboru programu, *-l* nám říká, že bude snort logovat do souboru jehož cesta je *var/log/snort.*

Příklad generovaného varování:

```
[**] [1:1122:5] WEB-MISC /etc/passwd [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/25-09:22:14.132982 10.0.0.2:2904 -> 192.168.1.2:80
TCP TTL:10 TOS:0x0 ID:34941 IpLen:20 DgmLen:82
```

Význam jednotlivých polí:

[**] - Označuje začátek výstrahy

[1:1122:5] Snort ID(1:1122) - Identifikace typu varování, (5) = revize pravidla

WEB-MISC /etc/passwd - Popis signatury

Attempted Information Leak - Klasifikace

Priority: 2 - Priorita útoku (nabývá hodnot v rozmezí 1- 4, 4 = nejvíce závažný útok)

03/25-09:22:14.132982 - Čas útoku

10.0.0.2:2904 - Zdrojová adresa a port

192.168.1.2:80 - Cílová adresa a port

TCP - Použitý protokol

TTL:10 - Doba života (Time to Live)

TOS:0x0 - Typ služby (Type of service) – priorita paketu

ID:34941 - ID paketu (Packet Identification)

IpLen:20 - Velikost IP hlavičky (IP length)

DgmLen:82 - Celková velikost paketu

4.2 Komponenty IDS systému Snort

Snort se skládá ze čtyř komponent. Z jednotky paketového záchytu, ze zásuvného modulu preprocesoru, z detekční jednotky a z výstupního zásuvného modulu.

Obr.6: Komponenty IDS systému Snort [2]

4.2.1 Jednotka paketového záchytu

Pomocí knihovny libpcap zachytává provozní data ze síťového rozhraní, které pak dále posílá zásuvným modulům preprocesoru. [2]

4.2.2 Zásuvné moduly preprocesoru

Tyto moduly mají za úkol úpravu (normalizaci) paketů přijatých z jednotky paketového záchytu a vlastní předzpracování, předtím, než se paket dostane k detekční jednotce. [2]

4.2.3 Detekční jednotka

Detekční jednotka má za úkol porovnávat přijatá data uvnitř každého paketu, zda neobsahují zvláštní řetězec, či hodnotu, které by se shodovaly s pravidly obsaženými ve Snortu. Pokud je naleznen příslušný datový řetězec, zobrazí se výstraha a proběhne zápis do logu. Tato jednotka je časově náročný modul, záleží tedy na výkonu počítače, na počtu používaných pravidel a také na aktuálním zatížení sítě. [2]

4.2.4 Výstupní zásuvný modul

Tento modul umožňuje provádět následující akce s výstupními logy a alerty:

- Záznam do souboru
- Posílání SMTP trapů
- Záznam do syslogu
- Záznam do databáze
- XML výstup
- Modifikace konfigurací směrovačů a firewallů

4.3 Pravidla Snortu

Součástí programu je i sada pravidel. Ty jsou standartně uloženy v */etc/snort/rules*. Výhodou Snortu je i fakt, že si uživatel může vytvářet, přidávat a upravovat pravidla tak, aby je přizpůsobil přesně svým potřebám. Každé pravidlo se skládá z **hlavičky** a **voleb**. [4,7]

Obecný tvar pravidla:

akce protokol zdr_IP zdr_port směr cíl_IP cíl_port (volby)

Ukázka pravidla ze souboru /etc/snort/rules

```
alert udp $EXTERNAL_NET any -> $HOME_NET 2140 (msg:"BACKDOOR
DeepThroat 3.1 Connection attempt"; content:"00"; depth:2;
reference:mcafee,98574; reference:nessus,10053; classtype:misc-
activity; sid:1980; rev:4;)
```

Každý přicházející paket se porovná, zda souhlasí jeho zdrojová adresa, cílová adresa a porty s pravidly obsaženými ve Snortu, pokud ano, je vyvolána příslušná akce. V tomto případě je vyvolán alert, pokud je přicházející paket z vnější sítě, využívá protokolu UDP a směřuje ve vnitřní síti na port 2140. Tehdy je zobrazena výstraha „BACKDOOR DeepThroat 3.1 Connection attempt“, což značí pokus o připojení a je detekován jako Backdoor Deep Throat hrozba.

4.3.1 Hlavička

Hlavička obsahuje akci, která se má vykonat, použitý protokol, zdrojovou, cílovou adresu a jejich porty.

pass (předání) – ignoruje pakety

log (zaznamenání) – zaznam paketu

alert (výstraha) – vygeneruje výstrahu a daný paket zaznamená

activate (aktivace) – vygeneruje výstrahu a testuje následující pravidla

dynamic (dynamika) – je nečinný do doby, než je vyvolán některým pravidlem z akce aktivace, poté záznam paketu

drop (zahození) – zaznamená paket a přidá do iptables pravidlo o zahození paketu

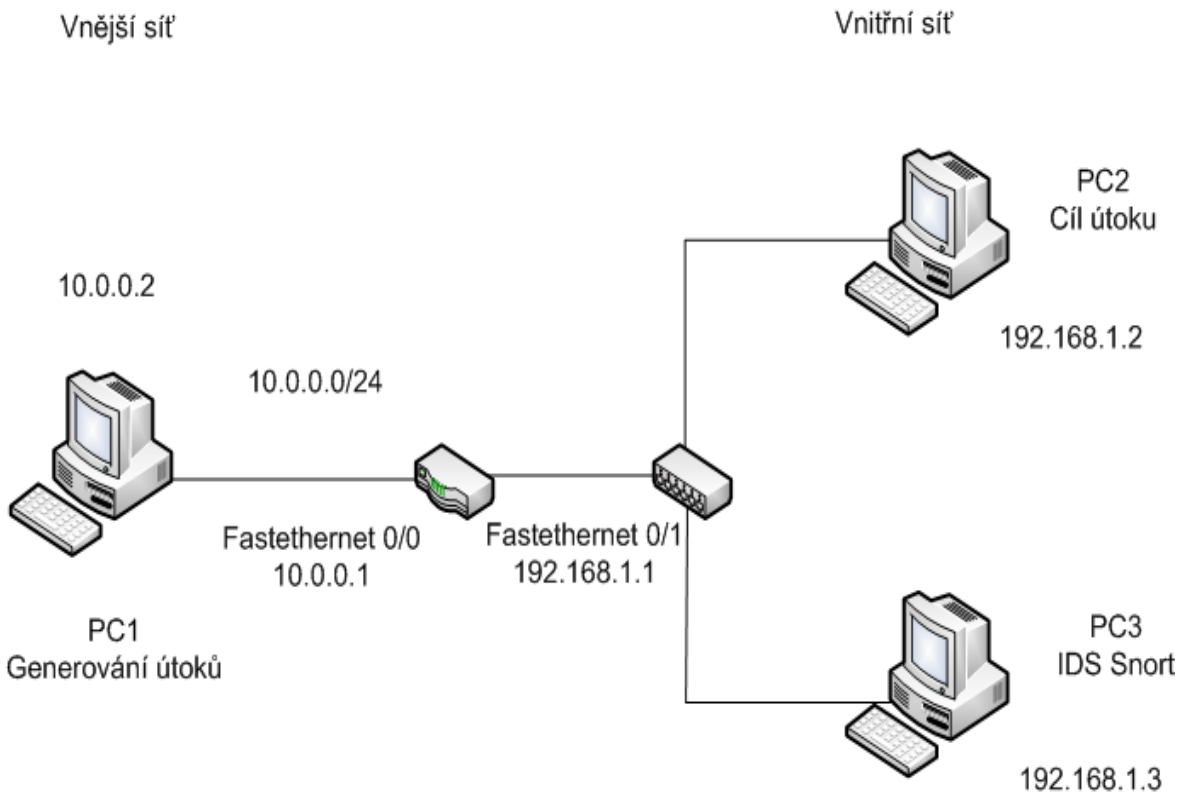
reject (odmítnutí) – přidá do iptables pravidlo pro zahození paketu, paket je zaznamenán a poslán TCP reset [4,7]

4.3.2 Volby

Volby umožňují kontrolu dalších atributů paketu a také vytvářejí popis k pravidlům. Msg: (textová informace), Reference (id systém), Gid (Generator ID), Sid (Snort rules ID), Rev (revize pravidla), Classtype (klasifikace pravidla), Priority (priorita pravidla), Logto (záznam), Content (prozkoumá paket, zda neobsahuje známý řetězec znaků), Nocase (bez rozlišování malých a velkých písmen), Offset (posun počáteční pozice pro hledání vzorku), Dsize (velikost paketu), Ttl (životnost paketu), Seq (sekvenční číslo). [4,7]

5 Návrh testovacího zapojení

Pro ověření zabezpečení jsem si zvolil tuto jednoduchou síť viz *Obr.7.* Skládá se ze tří počítačů, jednoho směrovače a jednoho rozbočovače. Pro měření jsem využil dostupnosti školní laboratoře N312. Počítač PC1 je umístěn ve vnější síti, což může reprezentovat síť Internet. Počítač PC2 je již umístěn ve vnitřní síti, která reprezentuje například interní síť firmy či zákazníka. Na dalším, posledním počítači, pojmenovaným jako PC3 je nainstalován IDS systém Snort, který je spolu s počítačem PC2 propojen přes rozbočovač (hub). Díky tomuto propojení je veškerý provoz směřující z vnější sítě na PC2 ve vnitřní síti kopírován i na PC3. V praxi se většinou setkáváme místo rozbočovače s přepínačem (switchem) s nakonfigurovaným SPAN portem. K adresaci jsem zvolil privátní IP adresy 10.0.0.0/24 a 192.168.1.0/24, jelikož počítače nejsou připojeny do sítě Internet. [1]



Obr.7: Schema testovacího zapojení [1]

5.1 Konfigurace zapojení

Počítač PC1 propojím křiženým kabelem se směrovačem (routerem) s rozhraním fastethernet 0/0, dále pak konzolovým kabelem, který mi umožní konfiguraci směrovače. Počítače PC2 a PC3 propojím přímým kabelem s rozbočovačem. Na PC1 spustím program Minicom, čímž se dostanu k nastavení směrovače. Pomocí následujících příkazů nastavím na rozhraní fastethernet 0/0, požadovanou IP adresu 10.0.0.1 a masku 255.255.255.0. [1]

Router>enable

Router#configure terminal

Router(config)#interface fastethernet 0/0

Router(config-if)#ip address 10.0.0.1 255.255.255.0

Router(config-if)#no shutdown

Obdobně nastavím i rozhraní fastethernet 0/1 pro připojení počítačů ve vnitřní síti:

Router>enable

Router#configure terminal

Router(config)#interface fastethernet 0/1

Router(config-if)#ip address ip address 192.168.1.1 255.255.255.0

Router(config-if)#no shutdown

Dalším krokem je nastavení IP adres a výchozí brány na všech počítačích - PC1, PC2 a PC3. Pro konfiguraci je nutné být přihlášen jako root, čímž si zajistím, mimo jiné, i možnost měnit síťová nastavení.

Po spuštění terminálu na každém z počítačů zadám následující posloupnosti příkazů:

Konfigurace pro PC1:

root@student-desktop:~# ifconfig eth0 10.0.0.2 netmask 255.255.255.0

root@student-desktop:~# route add default gateway 10.0.0.1

Konfigurace pro PC2:

root@student-desktop:~# ifconfig eth0 192.168.1.2 netmask 255.255.255.0

root@student-desktop:~# route add default gateway 192.168.1.1

Konfigurace pro PC3:

root@student-desktop:~# ifconfig eth0 192.168.1.3 netmask 255.255.255.0

root@student-desktop:~# route add default gateway 192.168.1.1

Pro ověření správného nastavení spustím terminál a vyzkouším ping z každého počítače na každý. Pokud mám vše dobře nakonfigurováno, měly by mi cílové stanice odpovědět.

root@ubuntu:~# ping 192.168.1.2

PING 192.168.1.2 (192.168.1.2) 56(84) bytes of data.

64 bytes from 192.168.1.2: icmp_seq=1 ttl=64 time=1.3 ms

64 bytes from 192.168.1.2: icmp_seq=2 ttl=64 time=1.3 ms

5.2 Instalace IDS systému Snort

Jako operační systém pro instalaci systému Snort byla zvolena linuxová distribuce Ubuntu 8.04 (Hardy Heron). Jelikož tento systém obsahuje Synaptic, což je grafický nástroj pro správu balíků založený na GTK+ a APT. Ten umožňuje instalovat, aktualizovat a odinstalovávat softwarové balíky v uživatelsky příjemném prostředí. Není nutné jednotlivé balíky rozbalovat, či kompilovat ze zdrojových kódů. Požadovaný software lze stáhnout a nainstalovat právě přes rozhraní Synapticu, díky němu se instalace podobá instalaci programů

v prostředí Windows. Po instalaci operačního systému jsem přidal tyto repozitory do souboru */etc/apt/sources.list.*

deb http://security.ubuntu.com/ubuntu hardy-security main restricted

deb-src http://security.ubuntu.com/ubuntu hardy-security main restricted

deb http://security.ubuntu.com/ubuntu hardy-security universe

deb-src http://security.ubuntu.com/ubuntu hardy-security universe

deb http://security.ubuntu.com/ubuntu hardy-security multiverse

deb-src http://security.ubuntu.com/ubuntu hardy-security multiverse

Poté se mi při vyhledání programu Snort pomocí Synapticu objevila možnost jeho instalace, včetně dalších potřebných součástí, kterými jsou libprelude2 (HIDS library), oinkmaster (správa a update Snort pravidel), snort-common, snort-common-libraries, snort-rules-default (pravidla pro Snort). V průběhu instalace se program zeptá, na kterém síťovém rozhraní chci, aby naslouchal. V mém případě jsem zvolil rozhraní *eth0.*

Obr.8: Nastavení rozhraní Snortu

5.2.1 Spuštění Snortu

Pro spuštění Snortu v režimu síťového IDS spustím terminál a zadám příkaz:

snort -c /etc/snort/snort.conf -A full -l /var/log/snort

Příkaz *snort -c /etc/snort/snort.conf* nám říká, jaká je cesta k uloženým pravidlům, přepínačem *-A full* nastavím ukládání alertů v plnohodnotném formátu, přepínač *-l /var/log/snort* uvádí cestu pro ukládání alertů. [7]

5.2.2 Konfigurace Snortu

Pro spuštění Snortu v režimu IDS je nejprve nutné upravit konfigurační soubor. Cesta k němu je defaultně */etc/snort/snort.conf.* [7]

5.2.2.1 Nastavení proměnných

Zde se konfigurují proměnné, které jsou použity v pravidlech. Je důležité nastavit, jaké IP adresy v síti budou reprezentovat vnitřní a jaké vnější síť. [7]

Nastavení pro vnitřní síť dostanu upravením řádku
var HOME_NET 192.168.1.0/24
Nastavení pro vnější síť provedu upravením řádku
var EXTERNAL_NET 10.0.0.0/24

Dále je možno nastavit adresy serverů DNS, SMTP, HTTP, SQL, TELNET a SNMP. V mém případě jsem ponechal defaultní nastavení pro všecny následující servery. Nastavení vypadá následovně:

var DNS_SERVERS $HOME_NET

var SMTP_SERVERS $HOME_NET

var HTTP_SERVERS $HOME_NET

var SQL_SERVERS $HOME_NET

var TELNET_SERVERS $HOME_NET

var SNMP_SERVERS $HOME_NET

5.2.2.2 Konfigurace dynamických knihoven

Ponechal jsem defaultní nastavení

dynamicpreprocessor directory /usr/lib/snort_dynamicpreprocessor/
dynamicengine /usr/lib/snort_dynamicengine/libsf_engine.so

5.2.2.3 Konfigurace preprocesorů

Preprocesory byly zavedeny již ve verzi Snortu 1.5. Jsou to moduly rozšiřující jeho funkčnost.

frag 3 - modul, zajišťující IP defragmentaci paketů, detekci útoků využívající IP defragmentaci.

preprocessor frag3_global: max_frags 65536
preprocessor frag3_engine: policy first detect_anomalies

stream4 - je modulem umožňující znovu sestavování TCP toku a umožňuje mu aplikovat stavovou analýzu. Díky tomuto je Snort schopen detekce stateless útoků.

preprocessor stream5_global: max_tcp 8192, track_tcp yes, \track_udp yes
preprocessor stream5_tcp: policy first, use_static_footprint_sizes

http_inspect - normalizuje a detekuje http provoz a protokolové anomálie.

```
preprocessor http_inspect: global \
   iis_unicode_map unicode.map 1252
preprocessor http_inspect_server: server default \
   profile all ports { 80 8080 8180 } oversize_dir_length 500
```

rpc_decode - normalizuje RPC provoz.

bo - Back Orifice detektor.

ftp_telnet - FTP & Telnet normalizér.

```
preprocessor ftp_telnet: global \
  encrypted_traffic yes \
  inspection_type stateful
```

*preprocessor ftp_telnet_protocol: telnet *
*normalize *
ayt_attack_thresh 200

smtp - dekodér pro uživatelské aplikace

ports { 25 }
*inspection_type stateful *
*normalize cmds *
*normalize_cmds { EXPN VRFY RCPT } *
*alt_max_command_line_len 260 { MAIL } *
*alt_max_command_line_len 300 { RCPT } *
*alt_max_command_line_len 500 { HELP HELO ETRN } *
alt_max_command_line_len 255 { EXPN VRFY }

sfPortscan - detekce různých typů portscanů. Nastaveno pro detekci TCP, UDP, ICMP, IP protokolů (*proto { all })*, maximální počet bytů pro alokaci (*memcap { 10000000 })*, citlivost nastavená na minimum.

*preprocessor sfportscan: proto { all } *
*memcap { 10000000 } *
sense_level { low }

arp spoof preprocesor - dekóduje a detekuje ARP útoky, unicastové ARP požadavky, ARP mapování. Defaultně vypnutý.

ssh preprocesor - detekuje následující exploity: Gobbles, CRC 32,

Secure CRT, a Protocol Mismatch exploit. Defaultně je vypnutý.

dns preprocesor – dekódování DNS provozu a detekce zranitelnosti

*preprocessor dns: *
ports { 53 }
enable_rdata_overflow

5.2.2.4 *Výstupní moduly*

Jednotný formát výstrah a logování. Budou vytvořeny dva soubory s názvem *snort.alert* a *snort.log*. Pro oba je nastavena maximální velikost souboru 128MB.

output alert_unified: snort.alert, limit 128
output log_unified: snort.log, limit 128

5.2.2.5 *Pravidla*

Snort využívá pravidla, která jsou uložena v textových souborech. Ta mohou být upravována uživatelem na základě potřeb. Pravidla jsou seskupena v kategoriích. Každá kategorie je uložena v samostatném souboru. Odkaz na tyto soubory je uveden v souboru *snort.conf*, který je načítán při každém spuštění Snortu v režimu IDS. Snort čte tyto pravidla a aplikuje je na zachycená data. Standardně jsou vypnuta pravidla pro detekci web-ataků, backdoorů, shellkódů, policy, porno, info, icmp-info, virus, chat, multimedia, a p2p. Pro použití i těchto pravidel je nutné odkomentovat příslušné řádky. [4]

6 Penetrační testování

Penetrační testy mají za úkol ověřit zabezpečení sítě. K tomuto účelu se využívají síťové nástroje jako bezpečnostní skenery, skenery portů, skenery spuštěných služeb, operačních systémů apod. Útok může být simulován jak z vnější sítě (většinou z internetu) do sítě vnitřní (podnikové), kde bývají webové servery, DNS servery, uložiště dat apod, tak i z vnitřní sítě (útok zaměstnance na podnikovou síť...) Penetrační testy jsou důležitou součástí bezpečnostní analýzy. Mají za úkol prověřit zabezpečení systémů vůči napadení a současně objevit slabá místa, kterými může být systém napaden. Slabá místa v informačním systému jsou hackery trvale vyhledávána a používané systémy jsou testovány na možnosti napadení. Aby bylo možno čelit jejich útokům, je nutné velmi podrobně sledovat a testovat informační technologie podobným způsobem. [6, 7]

Existují tři druhy penetračních testů:

- **Black-box test -** tester nemá žádné znalosti týkající se testovaného systému
- **White-box test -** tester má kompletní znalosti síťové topologie. Je k dispozici kompletní síťový diagram se seznamem hostů a operačních systémů
- **Gray-box test -** tester simuluje zaměstnance ve vnitřní síti. Má vytvořeny přístupové účty je jim umožněn standardní přístup do sítě.

6.1 IDSwakeup

IDSwakeup je nástroj umožňující testování IDS systémů. Umí generovat falešné pozitivní útoky, pro svou práci využívá knihovny hping 2 a iwu. Tento program jsem nainstaloval na PC1 (generování útoků), na PC3 byl spuštěný program Snort v režimu IDS, který zaznamenával jednotlivé útoky. [8]

Spuštění programu pomocí příkazu:

IDSwakeup <zdrojová IP adresa> <cílová IP adresa> [počet opakování] [hodnota ttl]

V mém případě vypadá příkaz pro spuštění takto:

```
root@ubuntu2:~# idswakeup 10.0.0.2 192.168.1.2 1 10
[: 16: -eq: unexpected operator

-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-
-  IDSwakeup : false positive generator                -
-  Stephane Aubert                                     -
-  Hervďż˝ Schauer Consultants (c) 2000                  -
-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

  src_addr:10.0.0.2  dst_addr:192.168.1.2  nb:1   ttl:10

  sending : teardrop ...
  sending : land ...
  sending : get_phf ...
  sending : bind_version ...
  sending : get_phf_syn_ack_get ...
  sending : ping_of_death ...
  sending : syndrop ...
  sending : newtear ...
  sending : X11 ...
  sending : SMBnegprot ...
  sending : smtp_expn_root ...
  sending : finger_redirect ...
  sending : ftp_cwd_root ...
  sending : ftp_port ...
  sending : trin00_pong ...
  sending : back_orifice ........atd
```

Spuštění programu IDSwakeup z příkazové řádky a generování útoků. Zde je zobrazena pouze část příkazové řádky, kompletní sken je uveden v příloze.

Příklady generovaných varování:

[**] [1:1002:7] WEB-IIS cmd.exe access [**]

[Classification: Web Application Attack] [Priority: 1]

03/25-09:22:13.955333 10.0.0.2:1714 -> 192.168.1.2:80

TCP TTL:10 TOS:0x0 ID:44473 IpLen:20 DgmLen:69

AP Seq: 0x5AF227E5 Ack: 0x3974D840 Win: 0x200 TcpLen:20

Tento alert zobrazuje pokus o přístup k command line na web serveru.

[**] [1:336:10] FTP CWD ~root attempt [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
03/25-09:22:24.123707 10.0.0.2:2152 -> 192.168.1.2:21
TCP TTL:10 TOS:0x0 ID:18018 IpLen:20 DgmLen:49
AP Seq: 0x204B7907 Ack: 0x6388D754 Win: 0x200 TcpLen: 20[Xref =>
http://cve.mitre.org/cgi-bin/cvename.cgi?name=1999-0082][Xref =>
http://www.whitehats.com/info/IDS318]

Tato výstraha upozorňuje na pokus o získání práv roota na FTP serveru.

[**] [122:1:0] (portscan) TCP Portscan [**]
[Priority: 3]
03/25-09:22:27.107206 10.0.0.2 -> 192.168.1.2
PROTO:255 TTL:0 TOS:0x0 ID:1812 IpLen:20 DgmLen:161

Zde je zachyceno skenování portů protokolu TCP, které pochází z vnější sítě z adresy 10.0.0.2 do vnitřní sítě na adresu 192.168.1.2.

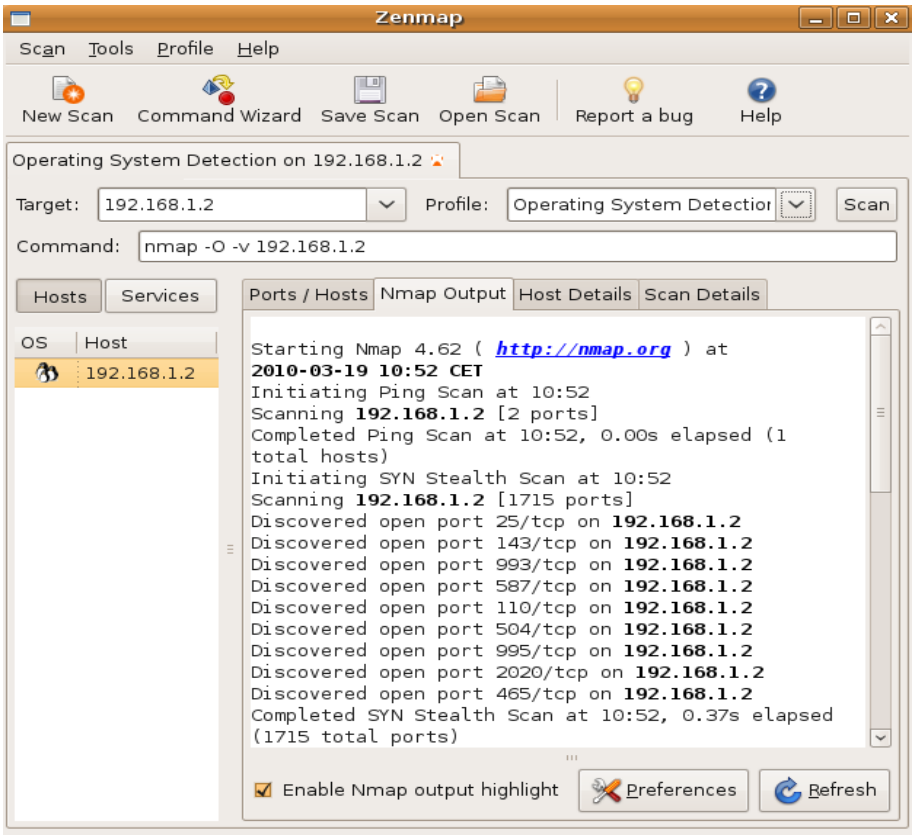
[**] [1:1002:7] WEB-IIS cmd.exe access [**]
[Classification: Web Application Attack] [Priority: 1]
03/25-09:22:13.955333 10.0.0.2:1714 -> 192.168.1.2:80
TCP TTL:10 TOS:0x0 ID:44473 IpLen:20 DgmLen:69
AP Seq: 0x5AF227E5 Ack: 0x3974D840 Win: 0x200 TcpLen: 20

Pokus o přístup na příkazový řádek web serveru.

6.2 Nmap

Nmap (Network Mapper) je bezplatný nástroj s licencí open source, určený k bezpečnostním auditům. Je využíván ke skenování portů, detekci operačního systému, běžících služeb... Byl navržen pro rychlé skenování velkých sítí. Nmap je možné instalovat na operační systémy Linux, Microsoft Windows a Mac OS X. Kromě klasického použití pomocí příkazového řádku existuje i grafické uživatelské prostředí (Zenmap), kterého ve své práci využiji. Na PC1 jsem nainstaloval Nmap a Zenmap pomocí správce balíků Synaptics. Po instalaci je vytvořen zástupce pro spuštění v nabídce Aplikace - Internet - Zenmap. Po spuštění

programu je nutné vybrat IP adresu počítače, který bude skenován. V mém případě se jednalo o PC2 s IP adresou 192.168.1.2. Dále pak si vyberu, jaký test chci provést a to pod záložkou Profile. [9]



Obr.9: Okno programu Zenmap

Generované výstrahy programu Snort při použití nástroje Zenmap:

[**] [122:1:0] (portscan) TCP Portscan [**]

[Priority: 3]

03/25-09:25:32.957080 10.0.0.2 -> 192.168.1.2

PROTO:255 TTL:0 TOS:0x0 ID:0 IpLen:20 DgmLen:157 DF

Zde je zachyceno skenování portů protokolu TCP, které pochází z vnější sítě z adresy 10.0.0.2 do vnitřní sítě na adresu 192.168.1.2.

[**] [116:59:1] (snort_decoder): Tcp Window Scale Option found with length > 14
[**][Priority: 3]
03/25-09:25:34.615107 10.0.0.2:62332 -> 192.168.1.2:1
TCP TTL:44 TOS:0x0 ID:12274 IpLen:20 DgmLen:60
U*PF Seq: 0x82AEB296 Ack: 0x2865221D Win: 0xFFFF TcpLen: 40 UrgPtr: 0x0
TCP Options (5) => WS: 15 NOP MSS: 265 TS: 4294967295 0 SackOK

Velikost okna specifikuje, kolik bytů dat se může přenést od odesílatele k příjemci bez průběžného potvrzování o doručení. Tato hodnota může nabývat hodnot v rozmezí 0-14. Zde je detekován paket s velikostí větší, než je povoleno.

[**] [1:1228:7] SCAN nmap XMAS [**]
[Classification: Attempted Information Leak] [Priority: 2]
03/25-09:25:34.615107 10.0.0.2:62332 -> 192.168.1.2:1
TCP TTL:44 TOS:0x0 ID:12274 IpLen:20 DgmLen:60
U*PF Seq: 0x82AEB296 Ack: 0x2865221D Win: 0xFFFF TcpLen: 40 UrgPtr: 0x0 TCP Options (5) => WS: 15 NOP MSS: 265 TS: 4294967295 0 SackOK
[Xref => http://www.whitehats.com/info/IDS30]

Tato výstraha ukazuje detekci skenování portů, typickou pro Nmap skener.

[**] [1:365:8] ICMP PING undefined code [**]
[Classification: Misc activity] [Priority: 3]
03/25-09:25:34.347241 10.0.0.2 -> 192.168.1.2
ICMP TTL:58 TOS:0x0 ID:34762 IpLen:20 DgmLen:148 DF
Type:8 Code:9 ID:28651   Seq:295 ECHO

Tento alert ukazuje na pokus o skenování, ping zahlcení, nebo zjištění, zda je skenovaná síť dostupná. Útok pochází z vnější do vnitřní sítě.

6.3 Nessus

Nessus je bezpečnostní skener, který slouží k analýze zranitelnosti hostů (sítí) a k nalezení potencionálních bezpečnostních rizik. Testy jsou implementovány pomocí zásuvných modulů (pluginů) mezi ně patří například:

- Backdoor pluginy
- Denial of Service (DoS)
- Skenování portů
- Skenování služeb
- Skenery web serverů
- Získání práv roota
- Web serverů a mnoho dalších

Nessus je možné pro domácí nekomerční užití používat zdarma, stačí se na webových stránkách www.nessus.org zaregistrovat, poté nám všechny potřebné informace dorazí na zadanou emailovou adresu i s návodem, jak produkt aktivovat. Je možné ho instalovat na operační systémy Microsoft Windows, Mac OS X, Linux, FreeBSD a Solaris.

Nessus je založen na architektuře klient - server. První částí je klient, který slouží k administraci, spouštění skenů, druhou částí je serverová část (nessus démon), který realizuje všechny bezpečnostní testy. Výhodou tohto rozdělení je možnost spouštění testování z kteréhokoliv místa v síti, přičemž nessus démon může běžet fyzicky na jiném počítači. [10]

6.3.1 Instalace Nessusu

Nejprve je nutné z webových stránek http://nessus.org/download/ stáhnut správnou verzi pro instalaci. V mém případě se jedná o Nessus verze 4.2 pro 32- bitový operační systém Ubuntu 8.04 (Nessus-4.2-ubuntu804_i386.deb), dále pak je nutné stáhnout Nessus klienta (NessusClient 4.0.2), který je určen pro Linuxové systémy. Jelikož mají oba balíčky příponou .deb (balíčkovací systém Debain), lze je snadno instalovat, podobně jako aplikace v prostředí Microsoft Windows. [10]

6.3.2 Konfigurace Nessusu

Nyní mám obě části programu nainstalovány, přistoupím ke konfiguraci. Spustím si terminál a pomocí následujících příkazů ho nastavím.

root@ubuntu:~# /etc/init.d/nessusd start

 # spuštění démona

$Starting Nessus : .
root@ubuntu:~# Missing plugins. Attempting a plugin update...
Your installation is missing plugins. Please register and try again.
To register, please visit http://www.nessus.org/register/

 # program se pokouší aktualizovat pluginy. Po registraci na webových stránkách produktu obdržíme aktivační kód, který pak vložíme zde:

/opt/nessus/bin/nessus-fetch --register E52F-CAAC-B3CE-3C27-B183

Your activation code has been registered properly - thank you.
Now fetching the newest plugin set from plugins.nessus.org...

 # aktivační kód byl přijat, nyní se stahují nejnovější pluginy

Your Nessus installation is now up-to-date.
If auto_update is set to 'yes' in nessusd.conf, Nessus will
update the plugins by itself.

 # nyní jsou všechny pluginy aktuální, je možné nastavit jejich automatickou aktualizaci

root@ubuntu:~# /opt/nessus/sbin/nessus-adduser
Login : admin
Login password :
Login password (again) :
Do you want this user to be a Nessus 'admin' user ? (can upload plugins, etc...) (y/n) [n]: y

 # nyní vytvoříme uživatele, který bude moci spouštět a konfigurovat testy

User rules
---
nessusd has a rules system which allows you to restrict the hosts
that admin has the right to test. For instance, you may want
him to be able to scan his own host only.

Please see the nessus-adduser manual for the rules syntax

Enter the rules for this user, and enter a BLANK LINE once you are done :
(the user can have an empty rules set)

Login : admin
Password : **********
This user will have 'admin' privileges within the Nessus server
Rules :

Is that ok ? (y/n) [y]
User added
root@ubuntu:~#

 # možnost přiřazení práv k uživateli

6.3.3 Spuštění programu

Pro spuštění programu je nutné, aby běžel nessus démon. To ověříme příkazem:

/etc/init.d/nessusd start

root@ubuntu:~# /etc/init.d/nessusd start
$Starting Nessus: nessus-service is already running as process 9225

 # služba nessus démon je spuštěna

Nessus démon tedy běží, můžeme přistoupit ke spuštění klientské části. Po instalaci se nám objeví v Aplikacích - Internet - NessusClient. Jeho spuštění je možné touto cestou, nebo pomocí webového prohlížeče otevřeme adresu, na kterém nessus démon běží, jako port zvolíme 8834 (**https://10.0.0.2:8834)**. Pro připojení je nutné použít protokol https, klasický http není podoprován kvůli bezpečnosti. Tímto se dostaneme do webového rozhraní programu. Nejprve je nutné se autentizovat. Použijeme údaje, které jsme zadávali při vytváření uživatele v předchozí kapitole. [10]

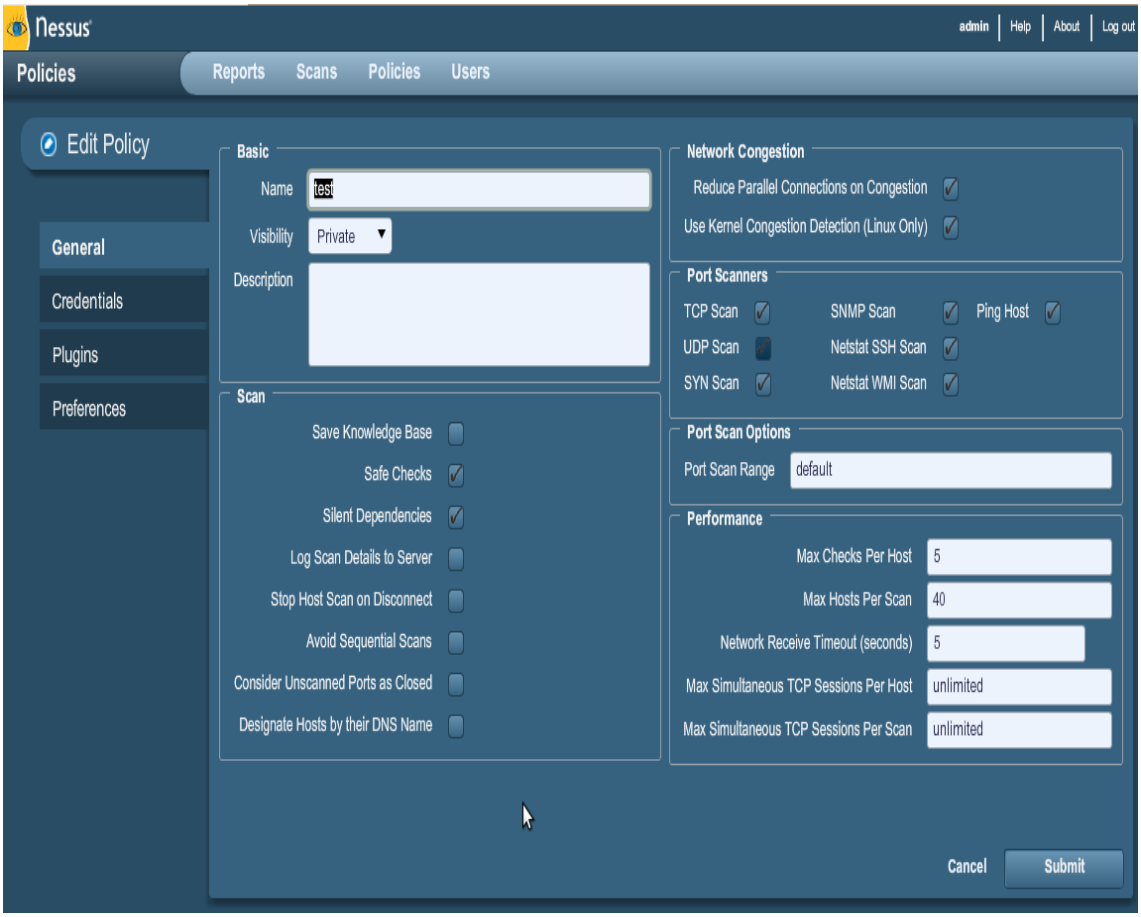
Obr. 10: Přihlášení klienta k serverové části (démonu)

Po úspěšném přihlášení se dostaneme ke konfiguraci programu. Základem jsou čtyři záložky:

- **Reports -** zde je možné nalézt uložené reporty z jednotlivých skenování
- **Scans -** tato záložka umožňuje přidávat a konfigurovat testy
- **Policies -** politika skenování, zde se nastavují parametry port skenů, použité pluginy, ochrana heslem…
- **Users -** uživatelé, kteří mají přístup do programu Nessus

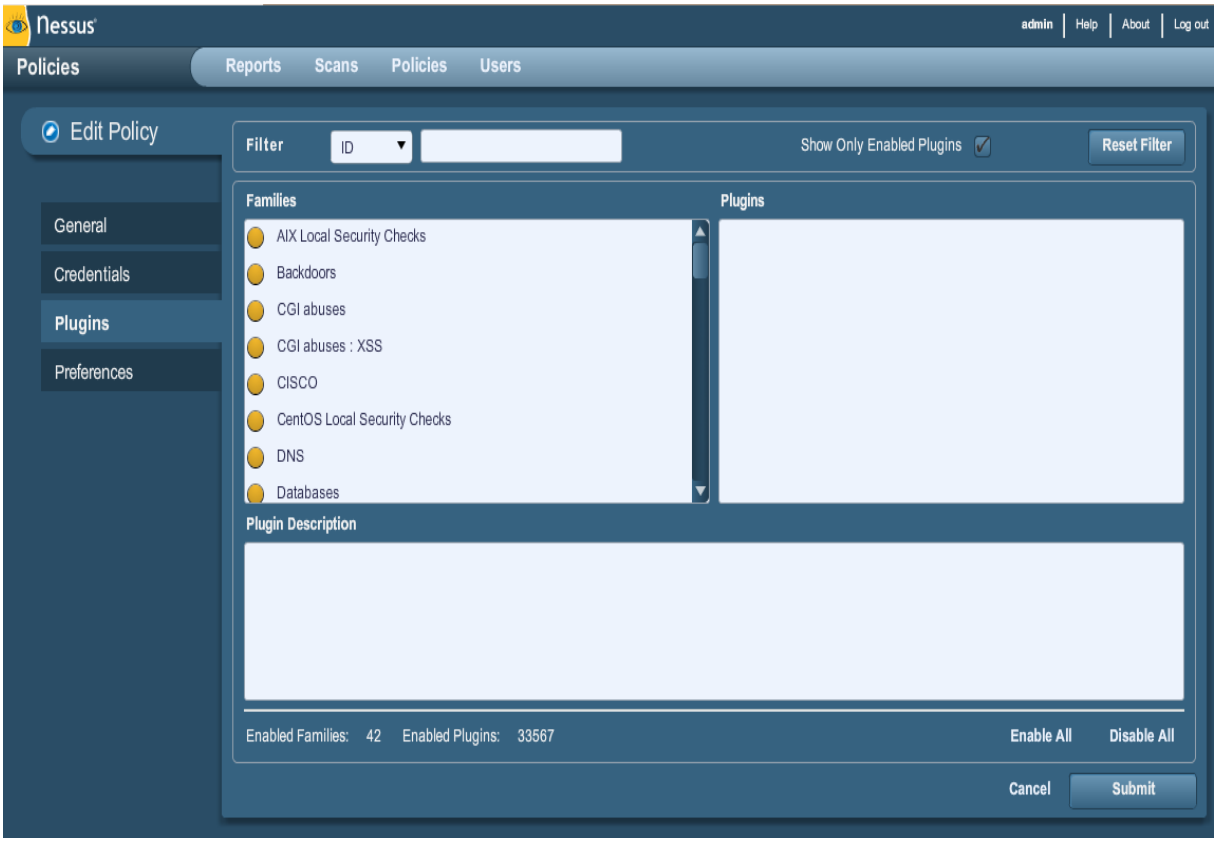
6.3.4 Nastavení pro test

Nejprve je nutné vytvořit a nastavit politiku pro skenování. Jako název si zvolím *test,* který nebude dostupný pro všechny uživatele (visibility - private), ale pouze pro uživatele, který ji vytvořil. Dále pak vyberu port skenery, které se mají při skenování použít. [10]



Obr. 11: Editování politiky skenu

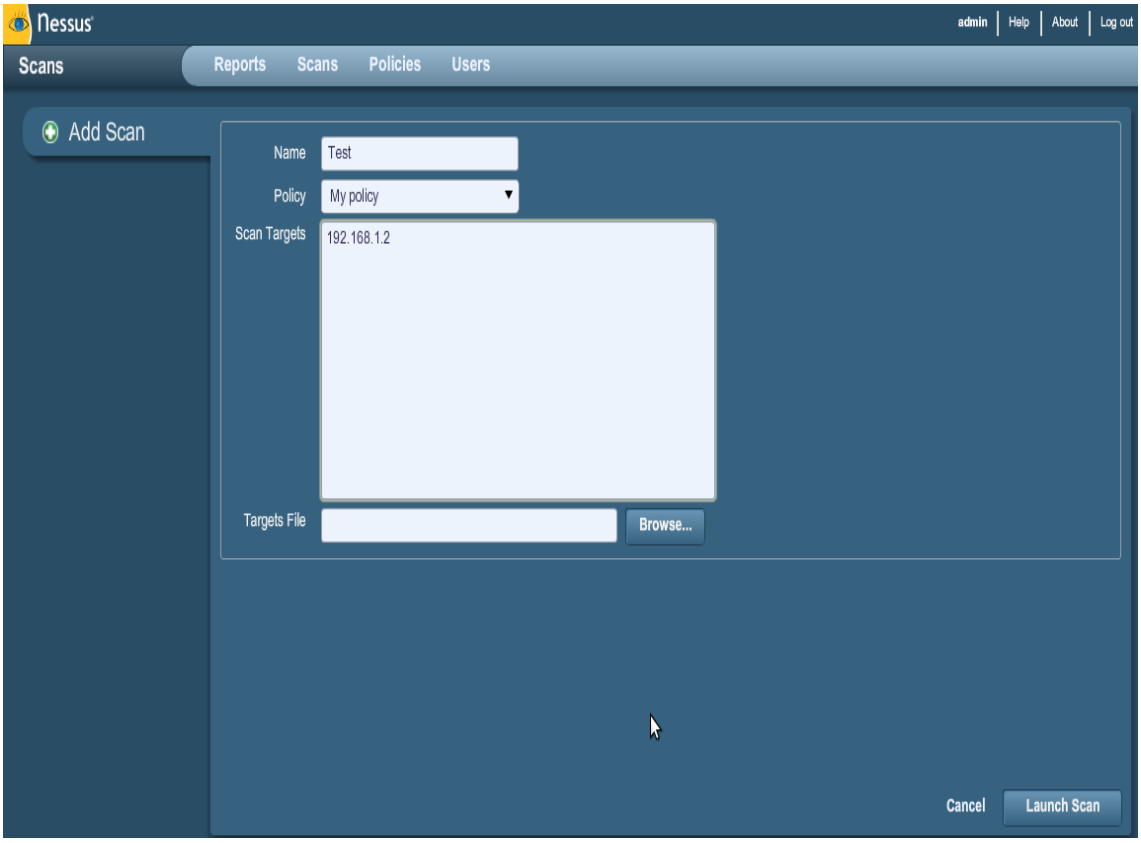
V dalším kroku je nutné vybrat pluginy, které budou při testu použity. Zvolím Enable All - tím zajistím, že cílový systém bude skenován všemi dostupnými pluginy. Výběr potvrdím stiskem Submit.



Obr. 12: Výběr pluginů pro otestování

6.3.5 Skenování

Nyní mohu přistoupit k vytvoření skenovacího plánu. Zvolím záložku Scan, vytvořím název testu, vyberu politiku pro sken a cílový stroj pro otestování. Skenování spustím tlačítkem Launch Scan. Nyní mohu na PC3, kde mám nainstalován a spuštěn Snort v režimu IDS sledovat generované výstrahy.



Obr.13: Vytvoření a spuštění skenu

6.3.6 Generované výstrahy programem Snort

[**] [116:45:1] (snort_decoder) TCP packet len is smaller than 20 bytes! [**][Priority: 3]
03/25-10:31:27.036367 10.0.0.2:0
-> 192.168.1.2:0 TCP TTL:64 TOS:0x0 ID:23472 IpLen:20 DgmLen:20

Zachycení paketu s hlavičkou menší než 20 bytů.

[**] [1:1444:3] TFTP Get [**]
[Classification: Potentially Bad Traffic] [Priority: 2]
03/25-10:26:58.230170 10.0.0.2:18010 -> 192.168.1.2:69
UDP TTL:64 TOS:0x0 ID:52756 IpLen:20 DgmLen:53
Len: 25

Trivial File Transfer Protocol (TFTP) umožňuje vzdáleným uživatelům kopírovat soubory bez jakéhokoli ověření. TFTP je někdy oprávněně používán pro zavedení image operačního systému, pro upload software, konfiguraci síťových prvků.

[**] [1:2049:4] MS-SQL ping attempt [**]
[Classification: Misc activity] [Priority: 3]
03/25-10:26:06.135839 10.0.0.2:52349 -> 192.168.1.2:1434
UDP TTL:64 TOS:0x0 ID:9280 IpLen:20 DgmLen:29 DF
Len: 1[Xref => http://cgi.nessus.org/plugins/dump.php3?id=10674]

Nessus tento ping používá k dotazu na existenci MS-SQL databáze běžící na počítači.

[**] [122:1:0] (portscan) UDP Portscan [**][Priority: 3]
03/25-09:30:38.56331 10.0.0.2 -> 192.168.1.2
PROTO:255 TTL:0 TOS:0x0 ID:0 IpLen:20 DgmLen:157 DF

Ukázka zachycení skenování portů protokolu UDP, které přichází z vnější sítě z adresy 10.0.0.2 do vnitřní sítě na adresu 192.168.1.2.

[**] [1:524:8] BAD-TRAFFIC tcp port 0 traffic [**]
[Classification: Misc activity] [Priority: 3]
03/25-10:28:20.121340 10.0.0.2:26110 -> 192.168.1.2:0
TCP TTL:64 TOS:0x0 ID:52129 IpLen:20 DgmLen:40
******S* Seq: 0x60A0BCD0 Ack: 0x0 Win: 0x200 TcpLen: 20

Detekce příchozího spojení na portu 0. Tento port je rezervován a nevyužívá se.

[**] [1:384:5] ICMP PING [**]
[Classification: Misc activity] [Priority: 3]
03/25-10:28:05.011770 10.0.0.2 -> 192.168.1.2
ICMP TTL:64 TOS:0x0 ID:53781 IpLen:20 DgmLen:29
Type:8 Code:0 ID:1 Seq:1 ECHO

Zjištění pingu do vnitřní sítě. Slouží ke zjištění dostupnosti cílového stroje.

7 Závěr

Cílem mé diplomové práce bylo objasnit problematiku IDS/IPS systémů, navrhnout a implementovat detekční systém, dále pak pomocí různých penetračních nástrojů ověřit schopnost detekce útoků. V první části jsem čtenáře seznámil s IDS a IPS systémy, jejich typy, vnitřní strukturou, vlastnostmi, jejich výhodami a nevýhodami. Dále jsem se zmínil o architektuře IDS/IPS. V další části jsem podrobně popsal síťový IDS systém Snort, postup jeho instalace a konfigurace pro úspěšnou detekci útoků. Poté jsem uvedl schema testovacího zapojení a následně jeho konfiguraci. V poslední části jsem ověřil funkčnost navrženého zapojení pomocí třech penetračních nástrojů, sledoval jsem, zda je IDS systém schopen tyto útoky zachytit. V této práci jsou ukázány pouze některé zachycené hrozby, kompletní výstrahy jsou uvedeny v příloze. Jak je z uvedených výstah zřejmé, podařila se mi detekce útoků na cílový systém. Zachyceny byly útoky typu port skeny, DDoS útoky, backdoory, pokusy o získání administrátorských práv, pokusy o přístup na vzdálený server, zjišťování dostupnosti cílového stroje a další.

Počítačová bezpečnost je velmi důležitým faktorem. Velké množství lidí dnes využívá osobní počítač ke své práci. A právě naše osobní data jsou tím nejcennějším. Jejich cenu pochopíme většinou až tehdy, když ně příjdeme. Proto je nutné být neustále ve střehu. Ke zvýšení bezpečnosti je dobré učinit alespoň základní kroky. Ke vhodné obraně je nutné mít záplatovaný operační systém, aktualizovaný antivir a dobře nakonfigurovaný firewall. Další zásadou může být omezení práv uživatelům, čímž zajistím menší možnosti zásahů do hostitelských systémů. V neposlední řadě je nutné používat zdravý rozum. V prostředí velkých sítí je nutné k těmto opatřením také nasadit IDS a IPS systémy. Vypracováním této diplomové práce jsem se posunul o kus dále se svými znalostmi v oblasti zabezpečení sítí. Předtím jsem měl zkušenosti pouze se zabezpečením jednotlivých osobních počítačů, nyní jsem si je rozšířil i o IDS/IPS systémy, které bývají nasazovány ve firemním prostředí.

8 Literatura

[1] Roman Aprias: *Systémy detekce průniku v Linuxu*

Dostupný z URL: http://www.cs.vsb.cz/grygarek/SPS/projekty0405/IDS/ids.html#sit

[2] Carl Endorf, Eugene Schultz, Jim Mellander: *Intrusion Detection & Prevention*,

McGraw-Hill 2004, ISBN:0072229543

[3] Raven Alder, Jacob Babbin, Adam Doxtater: *Snort 2.1 Intrusion Detection, Second Edition*

Syngress Publishing, Inc. 2004, ISBN: 1-931836-04-3

[4] Andrew R. Baker, Joel Esler: *Snort Intrusion Detection and Prevention Toolkit*,

Syngress Publishing, Inc. 2007, ISBN-10: 1-59749-099-7

Dostupný z URL : http://i.iinfo.cz/r/kd/Intrusion_Detection_with_SNORT.pdf

[5] Martin Roesch: *SNORT Users Manual 2.8.5*,

Dostupný z URL: http://www.snort.org/assets/125/snort_manual-2_8_5_1.pdf

[6] Andrew Whitaker, Daniel P. Newman, *Penetration Testing and Network Defense*,

Cisco Press, ISBN: 1-58705-208-3

[7] Filip Weber, *Penetrační testy v bezpečnostní analýze informačního systemu*

Dostupný z URL: http://www.svetsiti.cz/view.asp?rubrika=Technologie&clanekID=309

[8] *IDSwakeup*

Dostupný z URL: http://www.hsc.fr/ressources/outils/idswakeup/

[9] *Nmap Install Guide*

Dostupný z URL: http://nmap.org/

[10] Tenable Network Security, *Nessus 4.2 Installation Guide, Nessus 4.2 User Guide*

Dostupný z URL: http://www.nessus.org/documentation/

9 Seznam příloh

Složka Snort - konfigurační soubor programu Snort - *snort.conf*

Složka IDSwakeup - alerty - *alert.txt*

logy - *snort-unified.log*

Spuštění programu - *idswakeup generovani utoku.txt*

Složka Nessus - alerty - *alert.txt*

logy - *snort-unified.log*

Složka Nmap - alerty - *alert.txt*

logy - *snort-unified.log*